ActiveNeRF: Learning where to See with Uncertainty Estimation

Xuran Pan[1], Zihang Lai[2†], Shiji Song[1], and Gao Huang[1✉]

[1] Tsinghua University, Beijing 100084, China
pxr18@mails.tsinghua.edu.cn {shijis, gaohuang}@tsinghua.edu.cn
[2] Carnegie Mellon University, Pennsylvania 15213, United States
zihangl@andrew.cmu.edu

Abstract. Recently, Neural Radiance Fields (NeRF) has shown promising performances on reconstructing 3D scenes and synthesizing novel views from a sparse set of 2D images. Albeit effective, the performance of NeRF is highly influenced by the quality of training samples. With limited posed images from the scene, NeRF fails to generalize well to novel views and may collapse to trivial solutions in unobserved regions. This makes NeRF impractical under resource-constrained scenarios. In this paper, we present a novel learning framework, *ActiveNeRF*, aiming to model a 3D scene with a constrained input budget. Specifically, we first incorporate uncertainty estimation into a NeRF model, which ensures robustness under few observations and provides an interpretation of how NeRF understands the scene. On this basis, we propose to supplement the existing training set with newly captured samples based on an active learning scheme. By evaluating the reduction of uncertainty given new inputs, we select the samples that bring the most information gain. In this way, the quality of novel view synthesis can be improved with minimal additional resources. Extensive experiments validate the performance of our model on both realistic and synthetic scenes, especially with scarcer training data.

Keywords: Active Learning, Neural Radiance Fields, Uncertainty Estimation

1 Introduction

The task of synthesizing novel views of a scene from a sparse set of images has earned broad research interest in recent years. With the advent of neural rendering techniques, Neural Radiance Fields (NeRF) [20] shows its potential on rendering photo-realistic images and inspires a new line of research [37,22,24]. Different from traditional Structure-from-Motion [1] or image-based rendering [28] approaches, NeRF models the emitted radiance values and volume densities

† Work done during an internship at Tsinghua University.
✉ Corresponding author.

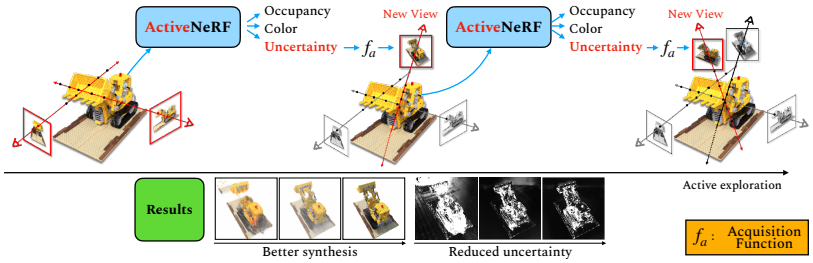

Fig. 1. ActiveNeRF: We present a flexible learning framework that *actively* expands the existing training set with newly captured samples based on an Active Learning scheme. ActiveNeRF incorporates uncertainty estimation into a NeRF model and evaluates the reduction of scene uncertainty at unobserved novel views. By selecting the view that brings the most information gain, the quality of novel view synthesis can be improved with minimal additional resources.

in a 3D scene as a function of continuous 5D coordinates, including spatial locations x, y, z and viewing directions θ, ϕ. The learned implicit function expresses a compact representation of the scene and enables free-viewpoint synthesis through volume rendering.

Despite its success in synthesizing high-quality images, the learning scheme for a NeRF model puts forward higher demands on the training data. First, NeRF usually requires a large number of posed images and is proved to generalize poorly with limited inputs [36]. Second, it takes a whole observation in the scene to train a well-generalized NeRF. As illustrated in Figure 2, if we remove observations of a particular part in the scene, NeRF fails to model the region and tends to collapse (*e.g.,* predicting zero density everywhere in the scene) instead of performing reasonable predictions. This poses challenges under real-world applications such as robot localization and mapping, where capturing training data can be costly, and perception of the entire scene is required [23,11,31].

In this paper, we focus on the context with constrained input image budget and attempt to address these limitations by leveraging the training data in the most efficient manner. As shown in Figure 1, we first introduce uncertainty estimation into the NeRF framework by modeling the radiance values of each location as a Gaussian distribution. This imposes the model to provide larger variances in the unobserved region instead of collapsing to a trivial solution. On this basis, we resort to the inspiration from active learning and propose to capture the most informative inputs as supplementary to the current training data. Specifically, given a hypothetical new input, we analyze the posterior distribution of the whole scene through Bayesian estimation, and use the subtraction of the variance from prior to posterior distribution as the information gain. This finally serves as the criterion for capturing new inputs, and thus raises the quality of synthesized views with minimal additional resources. Extensive experiments show that NeRF with uncertainty estimation achieves better performances on novel view synthesis, especially with scarce training data. Our

Fig. 2. Novel view synthesis of NeRF with partial observations. The models are trained with 10 posed images where observations from the left side are removed from the training set. While our model can still generate reasonably good synthesis results, the original NeRF shows large errors or completely fails to generate meaningful content.

proposed framework based on active learning, dubbed **ActiveNeRF**, also shows superior performances on both synthetic and realistic scenes, and outperforms several heuristic baselines.

2 Related Works

2.1 Novel View Synthesis

Synthesizing novel views of a 3D scene from a sparse set of 2D images is a long-standing problem in computer vision. Earlier work, including Structure-from-Motion [1] or image-based rendering [28], mostly reconstruct a scene in sparse representations. On this basis, bundle adjustment [33] and lighting-based approaches [15] consider the light and reluctance properties to synthesize photo-realistic images. More recently, the neural rendering technique has been introduced to the scene representation task, which inspires a line of research to model the 3D scene as a continuous representation. Scene Representation Network (SRN) [30] first models the scene as a function of 3D coordinates, which are then used to predict the intersections of object surfaces and the corresponding emitted color. Following SRN, Neural Radiance Fields (NeRF) [20] considers the volume density and view-dependent emitted color in the scene and models with a simple but effective multi-layer perceptron. The outputs in each location of the scene are combined with neural rendering techniques to synthesize novel views.

Many researches follow the step of NeRF and extend the original framework from different perspectives [35,2,14]. NeRF++ [37] analyzes the modeling capacity of NeRF and proposes an inverted sphere parameterization approach to model unbounded 3D scenes. FastNeRF [9] accelerates rendering procedure in NeRF to achieve real-time view synthesis. D-NeRF [24] and other related approaches propose to model dynamic scenes with moving objects. NeRF-W [18], on the other hand, focuses on modeling the transient objects varying from different images. Several works further extend NeRF-based models to represent scenes conditioned on a scene prior, which enables NeRF to generalize to new scenes.

More related to our work, several researches have also addressed the problem of NeRF under the limited input setting. Pixel-NeRF [36] proposes to encode the image-level features into the radiance field and trains a NeRF model that

can generalize across the scene. MVSNeRF [5] applies 3D CNN to reconstruct a neural encoding volume with per-voxel neural features. GRF [32] back-projects points to input images and gathers per-pixel features from each view. These approaches incorporate image features into original coordinate-based embeddings. DietNeRF [10] introduces additional semantic consistency loss with pretrained CLIP [25] models. Compared to these works, we are the first to address the limitation of NeRF from the data perspective and effectively increase the upper bound of the model with minimum additional resources. Also, the uncertainty estimation module in our framework is orthogonal to these approaches and can serve as a plug-and-play module to further boost their performances.

2.2 Uncertainty Estimation

The computer vision community has seen the value of uncertainty estimation in various research fields. Measuring the uncertainty of a neural network can both enhance the interpretability of the model outputs and reduce the risk of making critical faults. Based on the Bayesian rule, several approaches formulate uncertainty as a probability distribution over either the model parameters or model outputs. Bayesian Neural Networks (BNN) [17,13] approaches measure the uncertainty as posterior distribution, which usually require approximate inference methods, *e.g.,* variational inference. Dropout variational inference [7,12] estimates the model uncertainty with dropout layers in the network by performing multiple inferences for the same input.

Early research has also explored the possibility of applying uncertainty estimation in the field of novel view synthesis. NeRF-W [18] introduces uncertainty to model the transient objects in the scene. Compared to our approach, the uncertainty estimation in NeRF-W focuses on the differences across the images rather than the inherent noise inside the training data. Another concurrent work S-NeRF [27] models the uncertainty of the scene with variational inference. Although the uncertainty correlates well with the predictive error, S-NeRF performs qualitatively worse (*e.g.,* it shows blurry edges in the synthesis results) than the original model. It also requires multiple identical inferences to obtain the uncertainty map. Compared to these two approaches, our simple yet effective uncertainty estimation framework strictly follows the volume rendering procedure, and shows on par or better performances over the original NeRF model under various training data settings. The proposed uncertainty modeling is also a necessary component of the full ActiveNeRF framework: the uncertainty estimation serves as the basis to evaluate the new images.

2.3 Active Learning

Active learning has been widely studied in various computer vision tasks, including image classification [6], image captioning [21], and object detection [3]. Active learning can be categorized into two classes: representativeness-based and informativeness-based approaches. Representativeness methods rely on selecting examples by increasing the diversity of the training set. Core-set technique [26]

selects the samples by evaluating the Euclidean distance between candidates and labeled samples in the feature space. Also, several researches resort to the techniques in adversarial training [29] or self-supervised training [4], and select samples with an additional network, *e.g.,* a discriminator. More related to our work, informativeness methods measure the uncertainty of each data and select the most uncertain ones from an unlabelled data pool. With the uncertainty estimation approaches in the previous section, the selection criterion can be used in both Bayesian [8] and non-Bayesian [16] frameworks.

To the best of our knowledge, ActiveNeRF is the first approach to incorporating active learning scheme into the NeRF optimization pipeline. Unlike other works that focus on improving model capacities, we analyze the inherent imperfection of the training data, thereby increasing the synthesis quality of NeRF models with higher data efficiency. This is crucial for resource-constrained scenarios in real-world applications.

3 Background

In this section, we first briefly review the Neural Radiance Fields (NeRF) framework and introduce some implementation details.

NeRF models a scene as a continuous function F_θ which outputs emitted radiance value and volume density. Specifically, given a 3D position $\mathrm{x}=(x, y, z)$ in the scene and a viewing direction parameterized as a 3D Cartesian unit vector $\mathrm{d}=(d_x, d_y, d_z)$, a multi-layer perceptron model is adopted to produce the corresponding volume density σ and color $\mathrm{c}=(r, g, b)$ as follows:

$$[\sigma, f] = \mathrm{MLP}_{\theta_1}(\gamma_\mathrm{x}(\mathrm{x})), \tag{1}$$

$$c = \mathrm{MLP}_{\theta_2}(f, \gamma_\mathrm{d}(\mathrm{d})), \tag{2}$$

where $\gamma_\mathrm{x}(\cdot)$ and $\gamma_\mathrm{d}(\cdot)$ are the positional encoding functions, and f represents the intermediate feature independent from viewing direction d. An interesting observation is that the radiance color is only affected by its own 3D coordinates and the viewing direction, which makes it independent from other locations.

To achieve free view synthesis, NeRF renders the color of rays passing through the scene with the volume rendering technique. Let $\mathrm{r}(t) = \mathrm{o} + t\mathrm{d}$ be a camera ray with camera center $\mathrm{o} \in \mathcal{R}^3$ through a given pixel on the image plane, the color of the pixel can be formulated as:

$$C(\mathrm{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathrm{r}(t))\mathrm{c}(\mathrm{r}(t), \mathrm{d})ds, \tag{3}$$

where $T(t) = \exp(-\int_{t_n}^{t} \sigma(\mathrm{r}(s))ds)$ denotes the accumulated transmittance, and t_n and t_f are the near and far bounds in the scene. To make the rendering process tractable, NeRF approximates the integral based on stratified sampling, and formulates it as a linear combination of sampled points:

$$\hat{C}(\mathrm{r}) = \sum_{i=1}^{N_s} \alpha_i c(\mathrm{r}(t_i)),\ \alpha_i = \exp(-\sum_{j=1}^{i-1} \sigma_j \delta_j)(1 - \exp(-\sigma_i \delta_i)), \tag{4}$$

where $\delta_i = t_{i+1} - t_i$ is the distance between adjacent samples, and N_s denotes the number of samples. On this basis, NeRF optimizes the continuous function F_θ by minimizing the squared reconstruction errors between the ground truth from RGB images $\{\mathcal{I}_{i=1}^N\}$, and the rendered pixel colors.

To improve the sampling efficiency, NeRF optimizes two parallel networks simultaneously, and denote them as coarse and fine models respectively. The sampling strategy for the fine model is improved according to the result of the coarse model, where the samples are biased towards more relevant parts. In all, the optimization loss is parameterized as:

$$\sum_i \|C(\mathrm{r}_i) - \hat{C}^c(\mathrm{r}_i)\|_2^2 + \|C(\mathrm{r}_i) - \hat{C}^f(\mathrm{r}_i)\|_2^2, \tag{5}$$

where r_i is sampled ray, and $C(\mathrm{r}_i), \hat{C}^c(\mathrm{r}_i), \hat{C}^f(\mathrm{r}_i)$ correspond to the ground truth, coarse model prediction, and fine model prediction respectively.

4 NeRF with Uncertainty Estimation

In this paper, we focus on the context in some real-world applications, where the number of training data is within a limited budget. It has been proved in existing research [36] that NeRF fails to generalize well from one or few input views. If with incomplete scene observation, the original NeRF framework tends to collapse to trivial solutions by predicting the volume density as 0 for the unobserved regions.

As a remedy, we propose to model the emitted radiance value of each location in the scene as a Gaussian distribution instead of a single value. The predicted variance can serve as the reflection of the aleatoric uncertainty concerning a certain location. Through this, the model is imposed to provide larger variances in the unobserved region instead of collapsing to the trivial solution.

Specifically, we define the radiance color of a location $\mathrm{r}(t)$ follows a Gaussian distribution parameterized by mean $\bar{c}(\mathrm{r}(t))$ and variance $\bar{\beta}^2(\mathrm{r}(t))$. Following previous researches in Bayesian neural networks, we take the model output as the mean, and add an additional branch to the MLP network in Eq.(1) to model the variance as follows:

$$[\sigma, f, \beta^2(\mathrm{r}(t))] = \mathrm{MLP}_{\theta_1,\theta_3}(\gamma_\mathrm{x}(\mathrm{r}(t))), \tag{6}$$

$$\bar{c}(\mathrm{r}(t)) = \mathrm{MLP}_{\theta_2}(f, \gamma_\mathrm{d}(\mathrm{d})). \tag{7}$$

Softplus function is further adopted to produce a validate variance value:

$$\bar{\beta}^2(\mathrm{r}(t)) = \beta_0^2 + \log(1 + \exp(\beta^2(\mathrm{r}(t)))), \tag{8}$$

where β_0^2 ensures a minimum variance for all the locations.

In the rendering process, the new neural radiance field with uncertainty can be similarly performed through volume rendering. As we have mentioned in Sec. 3, the design paradigm in the NeRF framework provides two valuable prerequisites. (1) The radiance color of a particular position is only affected by its own 3D coordinates, which makes the distribution of different positions independent

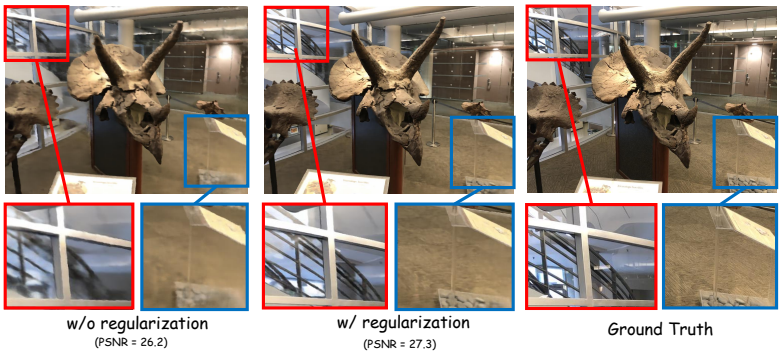

Fig. 3. Qualitative ablation on regularization term. The regularization term leads to more apparent synthesis results and greatly alleviates the blurs on the object surfaces. Quantitatively, regularization boosts performance by 1.1 PSNR on this scene

from each other. (2) Volume rendering can be approximated as linear combination of sampled points along the ray. On this basis, if we denote the Gaussian distribution of a position at $\mathrm{r}(t)$ as $c(\mathrm{r}(t)) \sim \mathcal{N}(\bar{c}(\mathrm{r}(t)), \bar{\beta}^2(\mathrm{r}(t)))$, the rendered value along this ray naturally follows Gaussian distribution:

$$\hat{C}(\mathrm{r}) \sim \mathcal{N}(\sum_{i=1}^{N_s} \alpha_i \bar{c}(\mathrm{r}(t_i)), \sum_{i=1}^{N_s} \alpha_i^2 \bar{\beta}^2(\mathrm{r}(t_i))) \sim \mathcal{N}(\bar{C}(\mathrm{r}), \bar{\beta}^2(\mathrm{r})), \tag{9}$$

where the α_is are the same as in Eq.(4), and $\bar{C}(\mathrm{r}), \bar{\beta}^2(\mathrm{r})$ denote the mean and variance of the rendered color through the sampled ray r.

To optimize our radiance field, we first assume that each location in the scene is at most sampled once in a training batch. We believe the hypothesis is reasonable as the intersection of two rays rarely happens in a 3D scene, let alone sampling at the same position in the same batch. Therefore, the distributions of rendered rays are assumed to be independent. In this way, we can optimize the model by minimizing the negative log-likelihood of rays $\{r_{i=1}^N\}$ from a batch $\mathcal{B}$:

$$\min_{\theta} \ -\log p_\theta(\mathcal{B}) = -\frac{1}{N}\sum_{i=1}^{N} \log\, p_\theta(C(\mathrm{r}_i)) = \frac{1}{N}\sum_{i=1}^{N} \frac{\|C(\mathrm{r}_i) - \bar{C}(\mathrm{r}_i)\|_2^2}{2\bar{\beta}^2(\mathrm{r}_i)} + \frac{\log\, \bar{\beta}^2(\mathrm{r}_i)}{2}. \tag{10}$$

However, simply minimizing the above objective function leads to a sub-optimal solution where the weights α_i for different samples in a ray are driven closer. This results in an unexpectedly large fraction of non-zero density in the whole scene, causing blurs on the object's surface, as depicted in Figure 3. Therefore, we add an additional regularization term to force sparser volume density, and the loss function is formulated as:

$$\mathcal{L}^{uct} = \frac{1}{N}\sum_{i=1}^{N} \left(\frac{\|C(\mathrm{r}_i) - \bar{C}(\mathrm{r}_i)\|_2^2}{2\bar{\beta}^2(\mathrm{r}_i)} + \frac{\log\, \bar{\beta}^2(\mathrm{r}_i)}{2} + \frac{\lambda}{N_s}\sum_{j=1}^{N_s} \sigma_i(\mathrm{r}_i(t_j)) \right), \tag{11}$$

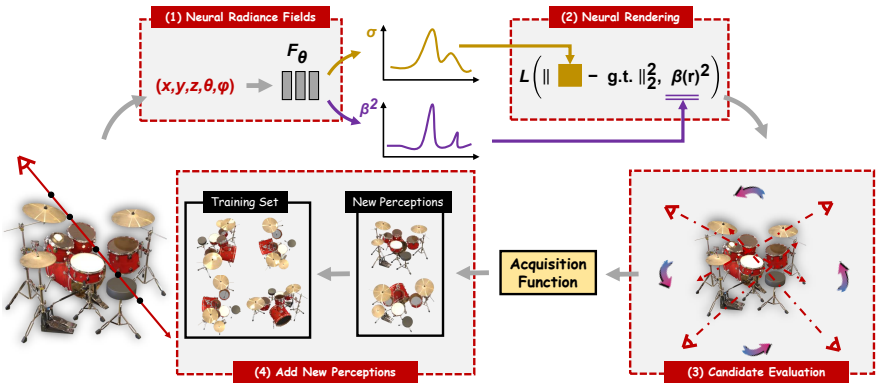

Fig. 4. The ActiveNeRF pipeline consists of 4 steps. First, the initial observation is used to train an ActiveNeRF model (Sec. 5.1). This model is then used to render novel views, from which the new viewpoint (that most reduces uncertainty) is estimated (Sec. 5.2 and 5.3). Finally, a new perception is captured and added to the training set

where λ is a hyper-parameter that controls the regularization strength.

We follow the original NeRF framework and optimize two parallel networks. To ease the difficulty of optimization, we only adopt the uncertainty branch in the fine model and keep the coarse model the same as vanilla. The final loss function is then:

$$\mathcal{L}^{uct}(C(\mathrm{r}), \bar{C}^f(\mathrm{r})) + \frac{1}{N}\sum_{i=1}^{N} \|C(\mathrm{r}_i) - \hat{C}^c(\mathrm{r}_i)\|_2^2. \tag{12}$$

By learning a neural radiance field as Gaussian distributions, we not only produce reasonable predictions in uncertain areas but also present an interpretation of how NeRF model understands the scene. On the one hand, uncertainty can be viewed as a perception of noises, which may also reflect the degree of risk in real-world scenarios, e.g., robotic navigation. On the other hand, this can further serve as a vital criterion in the following active learning framework.

5 ActiveNeRF

Although several works have attempted to model well-generalized NeRF under a limited training budget, the upper bound of their performances is highly restricted due to the inherent blind spot in the observations. For example, when modeling a car, if the right side of the car is never observed during training, the radiance field in this region would be under-optimized, making it almost impossible to render photo-realistic images.

Different from previous works, we target improving the upper bound of model performances. Inspired by the insights from active learning, we present a novel learning framework named ActiveNeRF and try to supplement the training sample in the most efficient manner, as illustrated in Figure 4. We first introduce

how to evaluate the effect of new inputs based on the uncertainty estimation and show two approaches for the framework to incorporate with new inputs.

5.1 Prior and Posterior Distribution

Estimating the influence of new data without its actual observation is a non-trivial problem. Nevertheless, modeling the radiance field as Gaussian distribution makes the evaluation more tractable, where we can estimate the posterior distribution of the radiance field based on the Bayesian rule.

Let D_1 denote the existing training set and F_θ denote the trained NeRF model given D_1. For simplicity, we first consider the influence of a single ray r_2 from the new input D_2. Thus, for the k_{th} sampled location $\mathrm{r}_2(t_k)$, its prior distribution is formulated as:

$$P^{(\mathrm{pri})} = P(c(\mathrm{r}_2(t_k))|D_1) \sim \mathcal{N}(\bar{c}(\mathrm{r}_2(t_k)), \bar{\beta}^2(\mathrm{r}_2(t_k))). \tag{13}$$

Following the sequential Bayesian formulation, the posterior distribution can then be derived as:

$$P^{(\mathrm{post})} = P(c(\mathrm{r}_2(t_k))|D_1, \mathrm{r}_2) = \frac{p(C(\mathrm{r}_2)|c(\mathrm{r}_2(t_k)))p(c(\mathrm{r}_2(t_k))|D_1)}{\int p(C(\mathrm{r}_2)|c(\mathrm{r}_2(t_k)))p(c(\mathrm{r}_2(t_k))|D_1)dc(\mathrm{r}_2(t_k))}. \tag{14}$$

As derived in Sec. 4, rendered color of rays follows the Gaussian distribution:

$$p(C(\mathrm{r}_2)|c(\mathrm{r}_2(t_k))) \sim \mathcal{N}(\bar{C}(\mathrm{r}_2), \bar{\beta}^2(\mathrm{r}_2)) \sim \mathcal{N}(\sum_{i=1}^{N_s} \alpha_i \bar{c}(\mathrm{r}_2(t_i)), \bar{\beta}^2(\mathrm{r}_2)). \tag{15}$$

As other sampled locations in r_2 are independent with $\mathrm{r}_2(t_k)$, we can represent the unrelated part in the mean as a constant $b(t_k)$ and the distribution can be simplified as:

$$p(C(\mathrm{r}_2)|c(\mathrm{r}_2(t_k))) \sim \mathcal{N}(\alpha_k \bar{c}(\mathrm{r}_2(t_k)) + b(t_k), \bar{\beta}^2(\mathrm{r}_2)). \tag{16}$$

Finally, by substituting terms in Eq.(14) with Eq.(13) and Eq.(16), the posterior distribution is formulated as:

$$P^{(\mathrm{post})} \sim \mathcal{N}\left(\gamma \frac{C(\mathrm{r}_2) - b(t_k)}{\alpha_k} + (1-\gamma)\bar{c}(\mathrm{r}_2(t_k)), \frac{\bar{\beta}^2(\mathrm{r}_2(t_k))\bar{\beta}^2(\mathrm{r}_2)}{\alpha_k^2 \bar{\beta}^2(\mathrm{r}_2(t_k)) + \bar{\beta}^2(\mathrm{r}_2)}\right), \tag{17}$$

$$\text{with} \quad \gamma = \frac{\alpha_k^2 \bar{\beta}^2(\mathrm{r}_2(t_k))}{\alpha_k^2 \bar{\beta}(\mathrm{r}_2(t_k)) + \bar{\beta}^2(\mathrm{r}_2)}. \tag{18}$$

Please refer to Appendix A for details.

5.2 Acquisition Function

With the posterior distribution formulated by the Bayesian rule, we quantitatively analyze the influence on the radiance field given a new input ray. As shown in Eq.(17), although the mean of the posterior distribution is unavailable due to the unknown of $C(\mathrm{r}_2)$, the variance is independent of the ground truth value and therefore can be *precisely computed* based on the current model F_θ. Additionally, it is worth noting that the variance of the posterior distribution of a

newly observed location $\mathrm{r}_2(t_k)$ is consistently smaller than its prior distribution:

$$\begin{aligned}\mathrm{Var}^{(\mathrm{post})}(\mathrm{r}_2(t_k)) &= \frac{\bar{\beta}^2(\mathrm{r}_2(t_k))\bar{\beta}^2(\mathrm{r}_2)}{\alpha_k^2\bar{\beta}^2(\mathrm{r}_2(t_k)) + \bar{\beta}^2(\mathrm{r}_2)} \\ &= (\frac{1}{\bar{\beta}^2(\mathrm{r}_2(t_k))} + \frac{\alpha_k^2}{\bar{\beta}^2(\mathrm{r}_2)})^{-1} < \bar{\beta}^2(\mathrm{r}_2(t_k)) = \mathrm{Var}^{(\mathrm{pri})}(\mathrm{r}_2(t_k)). \quad (19)\end{aligned}$$

This further proves that new observations can genuinely reduce the uncertainty of the radiance field. On this basis, we consider the reduction of variance as the estimation of information gain of $\mathrm{r}_2(t_k)$ from the new ray r_2:

$$\mathrm{Var}^{(\mathrm{pri})}(\mathrm{r}_2(t_k)) - \mathrm{Var}^{(\mathrm{post})}(\mathrm{r}_2(t_k)). \quad (20)$$

For a given image with resolution H, W, we can sample $N = H \times W$ independent rays, with N_s sampled locations from each ray. Therefore, we add up the reduction of variance from all these locations and define the acquisition function as:

$$\mathcal{A}(D_2) = \sum_{\mathrm{r}_i \in D_2} \sum_{j=1}^{N_s} \left(\mathrm{Var}^{(\mathrm{pri})}(\mathrm{r}_i(t_j)) - \mathrm{Var}^{(\mathrm{post})}(\mathrm{r}_i(t_j)) \right). \quad (21)$$

Similar derivation is also applicable with multiple input images, where the variance of posterior uncertainty is formulated as:

$$\mathrm{Var}^{(\mathrm{post})}(\mathrm{x}) = (\frac{1}{\bar{\beta}^2(\mathrm{x})} + \sum_i \frac{\alpha_{k_i}^2}{\bar{\beta}^2(\mathrm{r}_i(t_{k_i}))})^{-1}, \quad (22)$$

where r_i denotes ray from different images, and $\mathrm{x} = \mathrm{r}_i(t_{k_i}), \forall i$. Please refer to Appendix B for details.

In practical implementation, we first sample candidate views from a spherical space, and choose the top-k candidates that score highest in the acquisition function as the supplementary of the current training set. In this way, the captured new inputs bring the most information gain and promote the performance of the current model with the highest efficiency.

Besides, a quality-efficiency trade-off can also be achieved by evaluating new inputs with lower resolution. For example, instead of using full image size $H \times W$ as new rays, we can sample $H/r \times W/r$ rays to approximate the influence of the whole image with only $1/r^2$ time consumption.

5.3 Optimization and Inference

With the newly captured samples chosen by the acquisition function, we provide two approaches to incorporate the current NeRF model with additional inputs.
Bayesian Estimation. With the ground-truth value $C(\mathrm{r})$ from the new inputs, we can practically compute the posterior distribution of the locations in the scene by leveraging Eq.(17). Among these, the mean of distribution becomes the Bayesian estimation of emitted radiance value, and can be adopted in the

Table 1. Quantitative results in Fixed Training Set setting: ActiveNeRF performs superior to or on par with the original NeRF in all settings. In particular, note our model performs significantly better than NeRF in low-shot settings. We report PSNR/SSIM (higher is better) and LPIPS (lower is better)

	(a) Synthetic Scenes			(b) Realistic Scenes		
Method	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Setting I, training with all images						
SRN	22.26	0.846	0.170	22.84	0.668	0.378
LLFF	24.88	0.911	0.114	24.13	0.798	**0.212**
NeRF	**31.01**	0.947	0.081	**26.50**	0.811	0.250
IBRNet	25.62	0.939	0.110	-	-	-
MSVNeRF	27.07	0.931	0.168	-	-	-
Ours	30.45	**0.954**	**0.072**	25.96	**0.835**	**0.213**
Setting II, training with 10 images						
NeRF	28.04	0.866	0.134	23.36	0.791	0.280
DietNeRF	28.42	0.891	**0.087**	-	-	-
Ours	**28.51**	**0.932**	0.090	**23.96**	**0.803**	**0.260**
Setting III, training with 5 images						
NeRF	21.14	0.835	0.192	21.67	0.689	0.350
Ours	**23.23**	**0.866**	**0.185**	**22.03**	**0.712**	**0.292**

rendering process. At inference time, we only need to substitute the prior color with the posterior Bayesian estimation:

$$\bar{c}(\mathrm{r}(t_k)) \Rightarrow \gamma \frac{C(\mathrm{r}) - b(t_k)}{\alpha_k} + (1-\gamma)\bar{c}(\mathrm{r}(t_k)), \tag{23}$$

while others remain unchanged.

One of the advantages of using Bayesian estimation is that we avoid the collateral training procedure. If we consider an edge device, *e.g.,* a robot, the training-free scheme allows the agent to perform offline inference instantly, which is more friendly in resource-constrained scenarios.

Continuous Learning can also be considered if time and computation resources are not the bottlenecks. The captured inputs can be added to the training set and tune the model on the basis of the current one. We can further control the fraction of training rays from new images, forcing the model to optimize in the newly observed regions.

The two approaches can both promote the quality of the neural radiance field, and naturally achieve a trade-off between efficiency and synthesis quality.

6 Experiments

6.1 Experimental setup

Datasets. We extensively demonstrate our approach in two benchmarks, including LLFF [19] and NeRF [20] datasets. LLFF is a real-world dataset consisting of

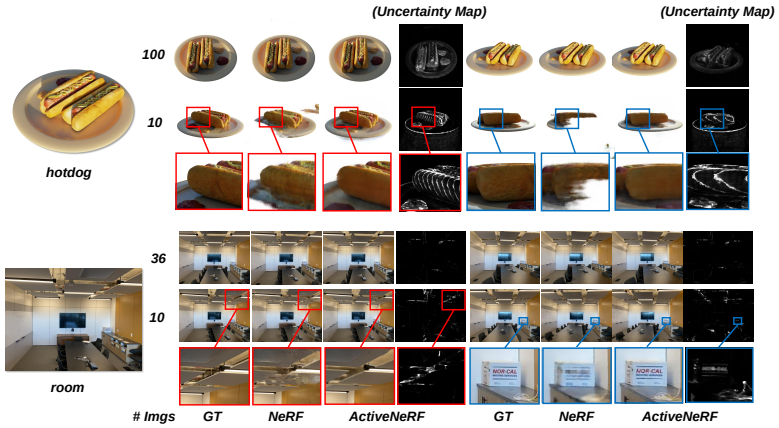

Fig. 5. Qualitative results on synthetic and realistic scenes with different fractions of training samples. Several observations can be made: First, ActiveNeRF performs significantly better than NeRF in the low-shot setting (*e.g.,* See Ln. 2 and 3). Second, the uncertainty correctly reduces when more data is used (See Col. *Uncertainty Map*). Finally, ActiveNeRF and NeRF obtain similar qualitative performance when all images are used (See Ln. 1 and 4), suggesting modeling uncertainty has no negative impact on the quality of view synthesis

8 complex scenes captured with a cellphone. Each scene contains 20-62 images with 1008×756 resolution, where 1/8 images are reserved for the test. NeRF dataset contains 8 synthetic objects with complicated geometry and realistic non-Lambertian materials. Each scene has 100 views for training and 200 for the test, and all the images are at 800×800 resolution. See detailed training configurations in the Appendix.

Metrics. We report the image quality metrics PSNR and SSIM for evaluations. We also include LPIPS [38], which more accurately reflects human perception.

6.2 Results

Uncertainty Estimation. We first evaluate the effectiveness of the proposed uncertainty estimation with different fractions of input samples. We compare with several competitive baselines, including Neural Radiance Fields (NeRF) [35], Local Light Field Fusion (LLFF) [19], and Scene Representation Networks (SRN) [30]. We also compare with three competitive baselines, including IBRNet [34], MSVNeRF [5] and DietNeRF [10].

We show the performance of our proposed approach with a different number of training data over baseline approaches in Table 1. It can be seen that NeRF with uncertainty performs on par or slightly better than baseline models, showing that modeling uncertainty does not affect the quality of synthesizing novel views. When it comes to limited training samples, our model shows consistently better results. For example, in the synthetic dataset, NeRF with 10% training data fails

Table 2. Quantitative results in Active Learning settings: BE: Bayesian estimation; **CL:** Continuous Learning; **Setting I:** 4 initial observations and 4 extra observations obtained at 40K, 80K, 120K, and 160K iterations. **Setting II:** 2 initial observations and 2 extra observations are obtained at 40K, 80K, 120K, and 160K iterations. **NeRF**†: NeRF performance from fixed training set setting. This setting measures NeRF's upper-bound performance by removing the difficulties introduced by continuous learning. Overall, ActiveNeRF outperforms baseline methods; several metrics could even match non-CL performance. We also report the total time consumption (training + inference time) of different approaches in the **Time** column, where **ActiveNeRF-BE** only consume training time at first 40K iterations and inference time at later stages

		(a) Synthetic Scenes			**(b) Realistic Scenes**		
Method	Time	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Setting I, 20 total observations:							
NeRF+Rand	2.0h	24.25	0.734	0.207	20.65	0.532	0.312
NeRF+FVS	2.0h	26.00	0.812	0.144	22.41	0.710	0.299
ActiveNeRF-BE	30min	25.67	0.778	0.169	21.86	0.644	0.303
ActiveNeRF-CL	2.2h	**26.24**	**0.856**	**0.124**	**23.12**	**0.765**	**0.292**
NeRF†	2.0h	28.04	0.910	0.134	23.36	0.791	0.280
Setting II, 10 total observations:							
NeRF+Rand	1.0h	18.36	0.642	0.251	18.49	0.478	0.355
NeRF+FVS	1.0h	19.24	0.735	0.227	20.02	0.633	0.344
ActiveNeRF-BE	16min	18.25	0.611	0.256	18.67	0.451	0.367
ActiveNeRF-CL	1.1h	**20.01**	**0.832**	**0.204**	**20.14**	**0.664**	**0.325**
NeRF†	1.0h	21.14	0.835	0.192	21.67	0.689	0.350

to generalize well to some views, while our model can still provide reasonable predictions. The gap is more distinct on the perceptual loss, *e.g.,* LPIPS, showing that our model can also render high-frequency textures with limited training data. Compared to DietNeRF, our model achieves better performances on two criteria and is competitive on the third. However, ours do not require additional pretrained model (*e.g.,* CLIP for DietNeRF) and can be used in the following active learning framework. Qualitative results are shown in Figure 5.

ActiveNeRF. We validate the performance of our proposed framework, ActiveNeRF, and compare it with two heuristic approaches. As an approximation, we hold out a large fraction of images in the training set and use these images as candidate samples. For baselines, we denote *NeRF+Random* as randomly capturing new images in the candidates. *NeRF+FVS (furthest view sampling)* corresponds to finding the candidates with the most distanced camera position compared with the current training set. We empirically adjust the number of the initial training set and captured samples during the training procedure.

We first show the results with continuous learning scheme, where the time and computation resources are considered sufficient. The comparison results are shown in Table 2 and Figure 6. We can easily see that ActiveNeRF captures the most informative inputs comparing with heuristic approaches, which contributes

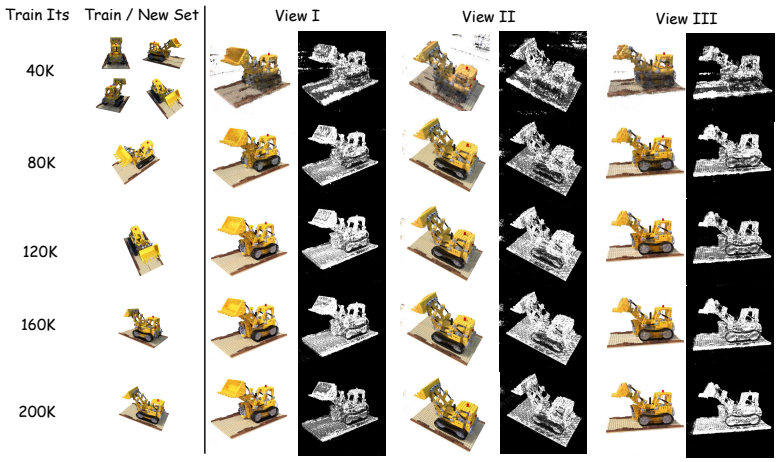

Fig. 6. Qualitative results of ActiveNeRF with four active iterations. We capture new perceptions every 40K iterations. Improved synthesis quality can be seen in unobserved regions

most to synthesizing views from less observed regions. The additional training cost for ActiveNeRF is also comparably minor (2.2h vs. 2h).

We further validate the model performances with Bayesian estimation. As shown in Table 2, 75% of the time consumption can be saved. Although showing inferior performance to continuous learning, the model with Bayesian estimation still synthesize reasonable images and is even competitive with heuristic approaches under continuous learning scheme.

7 Conclusion

In this paper, we present a flexible learning framework, that supplements the existing training set with newly captured samples based on an active learning scheme. We first incorporate uncertainty estimation into a NeRF model and evaluate the reduction of uncertainty in the scene given new inputs. By selecting the samples that bring the most information gain, the quality of novel view synthesis can be promoted with minimal additional resources. Also, our approach can be applied to various NeRF-extension approaches as a plug-in module, and enhance model performances in a resource-efficient manner.

Acknowledgement

This work is supported in part by National Key R&D Program of China (2021ZD0140407), the National Natural Science Foundation of China under Grants 62022048 and THU-Bosch JCML Center Beijing Academy of Artificial Intelligence.

References

1. Andrew, A.M.: Multiple view geometry in computer vision. Kybernetes (2001)
2. Arandjelović, R., Zisserman, A.: Nerf in detail: Learning to sample for view synthesis. arXiv preprint arXiv:2106.05264 (2021)
3. Bengar, J.Z., Gonzalez-Garcia, A., Villalonga, G., Raducanu, B., Aghdam, H.H., Mozerov, M., Lopez, A.M., van de Weijer, J.: Temporal coherence for active learning in videos. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW). pp. 914–923. IEEE (2019)
4. Bengar, J.Z., van de Weijer, J., Twardowski, B., Raducanu, B.: Reducing label effort: Self-supervised meets active learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1631–1639 (2021)
5. Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H.: Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14124–14133 (2021)
6. Fu, W., Wang, M., Hao, S., Wu, X.: Scalable active learning by approximated error reduction. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1396–1405 (2018)
7. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: international conference on machine learning. pp. 1050–1059. PMLR (2016)
8. Gal, Y., Islam, R., Ghahramani, Z.: Deep bayesian active learning with image data. In: International Conference on Machine Learning. pp. 1183–1192. PMLR (2017)
9. Garbin, S.J., Kowalski, M., Johnson, M., Shotton, J., Valentin, J.: Fastnerf: High-fidelity neural rendering at 200fps. arXiv preprint arXiv:2103.10380 (2021)
10. Jain, A., Tancik, M., Abbeel, P.: Putting nerf on a diet: Semantically consistent few-shot view synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5885–5894 (2021)
11. Khosoussi, K., Giamou, M., Sukhatme, G.S., Huang, S., Dissanayake, G., How, J.P.: Reliable graphs for slam. The International Journal of Robotics Research **38**(2-3), 260–298 (2019)
12. Kingma, D.P., Salimans, T., Welling, M.: Variational dropout and the local reparameterization trick. Advances in neural information processing systems **28**, 2575–2583 (2015)
13. Kononenko, I.: Bayesian neural networks. Biological Cybernetics **61**(5), 361–370 (1989)
14. Kosiorek, A.R., Strathmann, H., Zoran, D., Moreno, P., Schneider, R., Mokrá, S., Rezende, D.J.: Nerf-vae: A geometry aware 3d scene generative model. arXiv preprint arXiv:2104.00587 (2021)
15. Levoy, M., Hanrahan, P.: Light field rendering. In: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques. pp. 31–42 (1996)
16. Li, X., Guo, Y.: Adaptive active learning for image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 859–866 (2013)
17. MacKay, D.J.: Bayesian neural networks and density networks. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **354**(1), 73–80 (1995)

18. Martin-Brualla, R., Radwan, N., Sajjadi, M.S., Barron, J.T., Dosovitskiy, A., Duckworth, D.: Nerf in the wild: Neural radiance fields for unconstrained photo collections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7210–7219 (2021)
19. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. ACM Transactions on Graphics (TOG) **38**(4), 1–14 (2019)
20. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: European conference on computer vision. pp. 405–421. Springer (2020)
21. Miller, B., Kantchelian, A., Afroz, S., Bachwani, R., Dauber, E., Huang, L., Tschantz, M.C., Joseph, A.D., Tygar, J.D.: Adversarial active learning. In: Proceedings of the 2014 Workshop on Artificial Intelligent and Security Workshop. pp. 3–14 (2014)
22. Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., Martin-Brualla, R.: Deformable neural radiance fields. arXiv preprint arXiv:2011.12948 (2020)
23. Paull, L., Huang, G., Leonard, J.J.: A unified resource-constrained framework for graph slam. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). pp. 1346–1353. IEEE (2016)
24. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: D-nerf: Neural radiance fields for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10318–10327 (2021)
25. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning. pp. 8748–8763. PMLR (2021)
26. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. arXiv preprint arXiv:1708.00489 (2017)
27. Shen, J., Ruiz, A., Agudo, A., Moreno-Noguer, F.: Stochastic neural radiance fields: Quantifying uncertainty in implicit 3d representations. arXiv preprint arXiv:2109.02123 (2021)
28. Shum, H.Y., Chan, S.C., Kang, S.B.: Image-based rendering. Springer Science & Business Media (2008)
29. Sinha, S., Ebrahimi, S., Darrell, T.: Variational adversarial active learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5972–5981 (2019)
30. Sitzmann, V., Zollhöfer, M., Wetzstein, G.: Scene representation networks: Continuous 3d-structure-aware neural scene representations. arXiv preprint arXiv:1906.01618 (2019)
31. Torres-González, A., Martínez-de Dios, J.R., Ollero, A.: Robot-beacon distributed range-only slam for resource-constrained operation. Sensors **17**(4), 903 (2017)
32. Trevithick, A., Yang, B.: Grf: Learning a general radiance field for 3d representation and rendering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15182–15192 (2021)
33. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment—a modern synthesis. In: International workshop on vision algorithms. pp. 298–372. Springer (1999)

34. Wang, Q., Wang, Z., Genova, K., Srinivasan, P.P., Zhou, H., Barron, J.T., Martin-Brualla, R., Snavely, N., Funkhouser, T.: Ibrnet: Learning multi-view image-based rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4690–4699 (2021)
35. Wang, Z., Wu, S., Xie, W., Chen, M., Prisacariu, V.A.: Nerf–: Neural radiance fields without known camera parameters. arXiv preprint arXiv:2102.07064 (2021)
36. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelnerf: Neural radiance fields from one or few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4578–4587 (2021)
37. Zhang, K., Riegler, G., Snavely, N., Koltun, V.: Nerf++: Analyzing and improving neural radiance fields. arXiv preprint arXiv:2010.07492 (2020)
38. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)